INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING
Int. J. Optim. Civil Eng., 2021; 11(2): 291-327

GOLD RUSH OPTIMIZATION ALGORITHM

S. Sarjamei, M. S. Massoudi[*, †] and M. Esfandi Sarafraz
[1]*Islamic Azad University of West Tehran Branch, Civil Engineering Department, Tehran, Islamic Republic of Iran*

ABSTRACT

This article presents a new meta-heuristic optimization algorithm based on the power of human thinking and decision-making, which will be called Gold Rush Optimization (GRO). The thinking and decision-making ability of humans were used in this paper to develop a approach to create an optimization method. The hypothetical interaction between human operators in search of gold, based on the sound volume received from metal detectors, was used to develop the method. Benchmark functions, engineering design examples, and truss structures (which were optimized using different algorithms previously) were used for validation and verification of the proposed algorithm. MATLAB was used for programming. The CEC 2005 benchmark functions obtained reached the global target minimum, and the numerical engineering and truss examples were improved compared to the previous algorithms. Therefore, the proposed algorithm can be used as an alternative for the previously developed meta-heuristic optimization algorithms, which can be used in all optimization fields.

Keywords: Gold Rush algorithm; meta-heuristic optimal design; constrained optimization; human inspiration; GRO.

Received: 24 March 2021; Accepted: 28 May 2021

List of symbols
i the number of iteration
lb minimum allowable search space
ub maximum allowable search space
rand is a random number in the interval (0.1)
SOP is an operator who is successful in finding the optimal location
N is the number of variables – the number of dimensions in the search space

[*]Corresponding author: Department of Civil Engineering, Islamic Azad University of West Tehran Branch, Tehran, Iran.
[†]E-mail address: massoudi.ms@wtiau.ac.ir (M. S. Massoudi), https://orcid.org/0000-0002-8339-2470.

ρ parameter is described the fatigue effect
D coefficient of the distance
rate loudness of each sound
α probability of moving to the loudest sound or faring away from it
β probability of moving to the loudest sound or faring away from it
γ probability of moving to the loudest sound or faring away from it

1. INTRODUCTION

Meta-heuristic optimization techniques have become popular over the last decades. Different meta-heuristic methods have been proposed and used to solve complex problems such as engineering problems and solutions of truss structural. Meta-heuristic methods can be classified into three main categories: evolutionary (EA), physics-based, and Swarm Intelligence (SI) algorithms.

EAs are usually inspired by evolution in nature, and the most popular in this branch are Genetic Algorithms (GA). GA proposed by Holland [1], Goldberg [2] and used by Ding [3] and Liang [4]. GA is inspired by Darwin's theory about the biological evolutionary process. In general, optimization is achieved through the development of an initial random solution in EAs. New populations are formed through the combination and mutation of individuals in the previous population. Therefore, the best individuals have a higher probability of participating in creating the new population. Consequently, the new population is likely to be more superior than the previous generation(s), which means that the initial population is optimized throughout generations. Kooshkbaghi and Kaveh [5], introduced Artificial Coronary Circulation System (ACCS) algorithm, by inspired the growth of coronary arteries tree of the heart and coronary circulatory system in human beings, is applied to sizing optimization of truss structures. Some of the EAs are Genetic Programming (GP) Koza [6], and Biogeography-Based Optimizer (BBO) D. Simon [7], Differential Evolution (DE) Storn and Price [8], Evolutionary Programming (EP) Xin Yao [9] and Fogel [10], and Evolution Strategy (ES) Hansen [11], Rechenberg [12] and Kaveh [13].

The second main category of meta-heuristics is physics-based algorithms. Some algorithms in this category are Gravitational Local Search (GLSA) Webster [14], Kaveh and Ilchi Ghazaan [15] were produced Vibrating Particles System algorithm (VPS) inspired by free vibration of single degree of freedom systems with viscous damping. Also, they combined the vibrating particles system with multi-design variable configuration (MDVC-UVPS) Kaveh and Ilchi Ghazaan [16] to optimize the large scale space trusses. Big Bang–Big Crunch algorithm (BB–BC) formulated by Erol [17] and developed by Kaveh and Talatahari [18] have been formulated most recently, Gravitational Search Algorithm (GSA) proposed by Rashedi [19] is introduced using physical phenomena. Kaveh Kalateh-Ahani [20] introduced (CMA-ES) with Gaussian mutation (ES) and Covariance Matrix Adaptation (CMA) to achieve the optimized size of space trusses. Kaveh and Mahdavi [21], with laws of momentum and energy between collisions bodies, introduced the new algorithm, called Colliding Bodies Optimization (CBO). Enhanced Colliding Bodies Optimization (ECBO) introduced by Kaveh and Ilchi Ghazaan [22] improved the function of the CBO algorithm.

ECBO uses memory to save some best solutions. Kaveh and Talatahari [23] proposed a meta-heuristic algorithm called the charged system search (CSS). In order to explore the locations of the optimum, CSS uses the Coulomb and Gauss laws from physics and Newtonian laws from mechanics to guide the charged particles (CPs). Kaveh and Talatahari [24] used CSS for optimized truss problems. Kaveh and Motie Share [25] improved the function of the CSS algorithm and introduced Magnetic Charged System Search (MCSS). Kaveh and Mirzaei [26] used MCSS for optimized truss problems. Kaveh and Khayatazad [27], have introduced the Ray Optimization algorithm (RO) with dielectric materials and Snell's refraction law. Black Hole (BH) algorithm [28], Central Force Optimization (CFO) [29], Curved Space Optimization (CSO) algorithm [30], Small-World Optimization Algorithm (SWOA) [31], Artificial Chemical Reaction Optimization Algorithm (ACROA) [32], water evaporation optimization (WEO) [33] and Galaxy-based Search Algorithm (GBSA) [34] have a different mechanism than the EAs. These algorithms move throughout the search space based on physical rules, including gravitational force, ray casting, electromagnetic force, inertia force, weights.

The last subclass of meta-heuristics is the SI methods, which mostly follow the social behavior of swarms, herds, flocks, or schools of creatures in nature. The mechanism is very similar to physics-based algorithms, with the difference that search agents navigate using a simulation of creature collective and social intelligence. The most common SI technique is Particle Swarm Optimization (PSO). PSO formulated by Kennedy [35] simulates social situations and used by Kaucic [36]. The PSO algorithm uses multiple particles that chase the position of the optimal particle and their optimal positions obtained so far. In other words, a particle is moved based on its own best solution as well as the best solution obtained by the swarm. Gomes [37] has used of (PSO) algorithm as an optimization engine for reduced the mass of truss, Li LJ [38] have used heuristic particle swarm optimizer (HPSO), which is combined based on harmony search (HS) and standard particle swarm optimizer (PSO). Based on cuckoos' breeding behavior, Young introduced a new algorithm called Cuckoo Search (CS) Yang X-S [39], Gandomi [40] used CS for optimized truss problems. Ivan Zelinka and Michal Bukacek [41] focused on swarm intelligence techniques and its practical use in gold rush game. Tejani [42] for optimization, the size of space trusses has used Symbiotic Organisms Search (SOS), which is the biological interactions between organisms in an ecosystem. Benson Isaac and Douglas Allaire [43] to optimization of black-box models propose a gold rush (GR) policy that relies on purely local information to identify the next best design alternative to query. Mirjalili [44] adopted by hunting strategy of humpback whales, which called the Whale Optimization Algorithm (WOA). By dolphin sound, Kaveh and Farhoudi [45] introduced the Dolphin Echolocation (DE), which, as a sound strikes an object and sound-wave, is reflected towards it. The remaining so far proposed SI techniques are as follows: Ant Colony Optimization (ACO) inspired by Dorigo [46] patterned from the behavior of ant colonies. Marriage in Honey Bees Optimization Algorithm (MBO) Abbass [47], teaching-learning-based optimization (TLBO) [48,49], Harmony Search (HS) [50], Tabu (Taboo) Search (TS) [51], Group Search Optimizer (GSO) [52], Monkey Search [53], Dolphin Partner Optimization (DPO) [54], Bee Collecting Pollen Algorithm (BCPA) [55], Grey Wolf Optimizer [56].

This list shows that most of these algorithms are based on the rules governing the nature,

animal communication, and how they are hunted or find food. However, in this article, optimization is based on the thinking and decision-making of the human.

A gold rush is a discovery of gold sometimes accompanied by other precious metals and rare earth minerals that brings with it a burst of miners seeking their fortune. In Australia, New Zealand, Canada, Brazil, South Africa, and the United States, major gold rushes occurred in the 19th century, while there were smaller gold rushes elsewhere [57].

The objective of this research was to develop an approach in optimization using the thinking and decision-making superiority of humans to animals and nature. This study hypothesized that the afore-mentioned superior abilities of humans could improve the performance of the algorithm, hence reducing the benchmark function.

2. MATERIALS AND METHODS

2.1 Inspiration

In this research, an optimization algorithm, "Gold Rush Optimization Algorithm" was developed. The optimization algorithm was created based on the thinking and decision-making power of humans. A hypothetical situation was considered where a group of people was searching for gold. Each of the group members is called an "operator," and the group operates in a circumscribed space called "search space." Every operator was standing in a random spot using a metal detector in order to find gold, as shown in Fig. 1.

Figure 1. Random spread of the operators in the search space (Photo reproduced from https://www.nhpr.org/post/hunting-treasure-nh-s-largest-metal-detecting-event#stream/0)

A metal detector is an electronic device, which detects the presence of nearby metal that sounds present metal inclusions are hidden inside objects or metal objects buried underground. The sound generated by the metal detector changes according to the distance from metal objects. In other words, the sound gets louder when the device approaches a metal object and vice versa. The operator depending on the sound volume, could estimate the proximity of the buried gold (called "fitness"), and change their movement direction. In

every stage, the operators move altogether and listen to the sound until they hear an increase in the sound and then stop at that point.

Every operator would also listen to the sounds produced by other devices and constantly monitor if any other devices create a louder sound. Also, every operator must consider the sources of error around themselves and their search space including

1. The distance between the operators, and the more the distance, the sound is perceived less than the actual;
2. The fatigue of the operators which increases after a few stages;
3. Error in the type of metal found by the detector other than gold.

One of the cons of the proposed GRO algorithm is the false sound received from the metal detector once it detects some type of metal other than gold, for instance, a can. In order to mitigate this risk, the operators not only consider the loudest sound but a group of average sounds, and once faced with unwanted metal items, they gradually remove them from the field. In the process of hearing, the sound is transmitted from ears by the vestibulocochlear nerve to the brainstem.

Sensory cores analyze the information received from the brainstem. The analyzed information is transmitted to the upper parts of the brain called basal ganglia, which is the center for decision-making and emotion. Eventually, the brain makes a decision based on the detector sound, and one of the described errors may happen.

The Papez Circle is a decision-making process in the brain. The Papez Circle enters different cycles such as hippocampus nuclei, amygdala nuclei. At this stage, based on which cores are selected or bypassed, different decisions are made by the human.

The information goes to the thalamus, and brain cortex of the frontal lobe, respectively, that takes the final decision, as shown in. After the frontal lobe took the decision, it sends information through the fascicle to the parietal lobe, and the parietal lobe returns the information to basal ganglia. The position of the brain lobe, as shown in Fig. 2. Basal ganglia send the information from the lobes to the midbrain and ultimately to the spinal cord and organs, and then the move will take place [58].

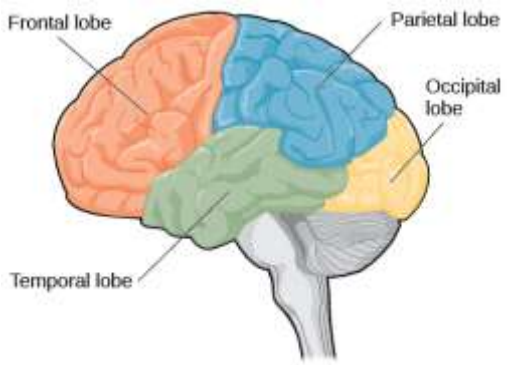

Figure 2. Different lobes of the brain cortex (reproduced from https://www.knowyourbody.net/parietal-lobe.html)

The advantage of the GRO algorithm is the selection of different movement options without a force in decision-making. In contrast, the algorithms that are based on the natural

laws or instinctual behavior of the animals entail some sort of force in decision-making. The brain analyzes the sounds and errors; according to that, the operator chooses the best direction as close as possible to gold. By using a team-work approach, the operators can get steadily closer to gold after only a few movements, which saves time and energy.

2.2 Presentation of gold rush optimization

Based on the above-mentioned hypothetical procedure, the GRO algorithm in the following equation is expressed. Finally, similar to the other metaheuristic approach, the proposed algorithm is an iterative process that starts by initializing the global parameters such that: For the location "i" lb_i and ub_i are the minima and the maximum allowable values. lb_i and ub_i are the search space of the operators. Rand in the interval (0.1) is a random number. As mentioned above, this algorithm is based on teamwork. Accordingly, for the best performance of the algorithm, more than one operator was selected. The size of the operators' population with 2 ≤operators. A successful operator (SOP) is an operator who is successful in finding the optimal location.

Each of the operators has its transfer restrictions, which are the constraints of the problem. Afterward, the main constants are calculated, and the following steps explain the primary transformations involved in the pathway of the algorithm.

Level 1: Initialization

Initially, a group of operators enters the search space. Every operator stands randomly in one spot within the search space (boundary condition) is described in Eq. (1), holding a metal detector.

$$location_i^{(0)} = lb_i + \left(ub_i - lb_i\right) * rand \quad , \quad i = 1, 2, \ldots, N \tag{1}$$

The N is the number of variables – the number of dimensions in the search space equals the number of variables. Each operator can move in N directions and find their next spot.

Every operator determines the success of people in finding the excellent location they are standing based on the loudness of the sound received from the metal detector. An operator can determine their distance to gold based on the sound loudness. This way, an operator selects their best movement direction to approach the gold, which is the goal of all the operator group.

Level 2: Monitoring-Choosing the best locations

In this step, SOP should be generated. In the first iteration, top 10 percent of operators should be selected and stored in SOP. In second or more iteration, SOP of previous iterations should be compared with population of operators and top 10 percent of union of them should be selected and stored as SOP.

Level 3: Fitness-distance

The evaluation of the loudness of each sound (rate), people with an excellent chance to find gold, is calculated based on Eq. (2):

$$\text{rate}(i) = \frac{D_i}{\rho_i} * \frac{sound\left(highest\,volume\right) - sound\left(i\right)}{\left(sound\left(highest\,volume\right) - sound\left(lowest\,volume\right) + \varepsilon\right)} \tag{2}$$

Considering the coefficients ρ_i moreover, D_i, the errors coming from the environment, operators are addressed, and the accuracy of the heard sound is increased. The epsilon (ε) is small positive number to avoid singularities.

Due to the frequent movements of the operators, they get tired gradually, which affects their search quality. In fact, in each movement stage of the group, the operators have less movement efficiency than the previous stages, and in each stage, their movement range is shortened. To overcome such a problem, we introduce a parameter ρ_i is described in Eq. (3) in order to measure the fatigue effect, which is the cause of poor quality detection. In fact ρ_i is a control parameter of the exploitation and the exploration process. Using the parameter ρ_i the rate of the movement of each operator in each stage is calculated.

$$\rho_i = 2 - \frac{iter}{max_{iter}} \tag{3}$$

The distance between the operators affects the perception of the sound heard by each operator. Due to the attenuation of sound in the air, the one with the longest distance is perceived less than actual. In order to calculate the attenuation of sound, the distance D_i between the operators in Eq. (4) is considered based on the coefficient of the distance. This way, the loudness of a sound can be accurately evaluated. The indices i and j indicate the current position of the two operators.

$$D_i = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 + \ldots} \tag{4}$$

Level 4: Think-Decisions-move

In this step, each operator can make different decisions based on a combination of the sounds. Each operator thinks and decides to move towards which direction based on the sound received from their own and others' metal detectors. Sometimes metals other than gold is found, and consequently, in order to mitigate the risk of capturing metals other than gold, the operators do not necessarily move towards the loudest sound. In other words, the operators may need to move towards or away from the loudest sound.

Each operator can move towards or against the loudest sound. Finally, the final result (location) is multiplied by Rate to get the updated location in Eq. (5).

$$new location\left(i\right) = location\left(i\right) + dm \times \left[\left(rate\left(j\right) - rate\left(i\right)\right) * \left(location\left(j\right) - location\left(i\right)\right) * rand\right] \tag{5}$$

Therefore, an operator does not always move towards the loudest sound but sometimes may move away from it. Therefore, a random number (between 0.0 and 1.0) is chosen, and

this number is compared with the ranges of α in Eq. (6).

$$md = \begin{cases} +1 \Rightarrow towards\ a\ loudest\ sound & \alpha > rand \\ -1 \Rightarrow away\ from\ a\ loudest\ sound & \alpha < rand \end{cases} \quad (6)$$

md means move direction. The first range ($\alpha > rand$) corresponds to moving towards the loudest sound, and the second range ($\alpha < rand$) corresponds to moving away from the loudest sound. Using this strategy in the GRO algorithm prevents getting trapped in a local optimum and instead select the answers by the movement towards a global optimum.

Level 5: Correct location

Also, the sound heard by the operator is analyzed in the frontal lobe of the brain, which makes different decisions. Considering the selection of different nuclei by basal ganglia in Papez Circle, a wide range of decisions are taken for the operator to move towards a specific direction. The decisions are finalized in the frontal lobe, and movement is made towards a direction.

If the location obtained in Eq. (5) violates the constraints of the problem, the Eq. (7) is used to create new locations. Therefore, a random number (between 0.0 and 1.0) is chosen, and this number is compared with the ranges of β and γ where $0 < \beta < \gamma < 1$. Each operator according to the above parameters selects one of the following activities:

$$new\ location(i) = \begin{cases} choose\ a\ neighboring\ location & rand < \beta \\ select\ a\ new\ location\ randomly & \beta < rand < \gamma \\ do\ not\ move & \gamma < rand \end{cases} \quad (7)$$

Interpolation is used to find a location in the neighborhood.

Level 6: Termination

Eventually, steps 4 to 6 are repeated in a loop until one of the terminating criteria as below is achieved:

1. The maximum number of allowed attempts (Max_{iter}) is 400.
2. No further change in the best location is observed (after several iterations without improvement).
3. The difference between the values of the SOP function and the global optimum is less than a pre-fixed anticipated threshold.
4. If the difference between the objective values of the best and the worst location becomes less than a specified accuracy.

The pseudo-code of the GRO algorithm is presented as follows.

Pseudo-code for the proposed algorithm GRO:

Initialize

Step 1. Initialize the gold rush algorithm parameters.

1-1: Number of operator (i=1,2,…,n)

1-2: Every operator stands randomly

Step 2. Determines the success of people in finding an excellent location.

Step 3. A number of the SOP are chosen and recorded.

Search
Step 4. Set the parameter $\alpha, \beta, \text{and } \gamma$
While (iter<Max number of iterations)
For each operator
For each operator + some of SOP
Each operator determines the following parameters: fatigue, distance, and rate.
Calculate the location of the current search operator by Eq. 5.
Allocate the probability of the moving towards or away from the loudest sound using the parameters $\alpha, \beta, \text{and } \gamma$.
Update locations.
End for
End for
Step 6.
Calculate the success of people in finding an excellent location (SOP).
Update the best location of all the operators.
End while
Memorize the best location of the operators.
Step 7. If not (terminate) iter=iter+1; go to Step 4;

The algorithm uses the number of function evaluations equal to (number of population + number of SOPs)*number of iterations.

The flowchart of the GRO algorithm is illustrated in Fig. 3.

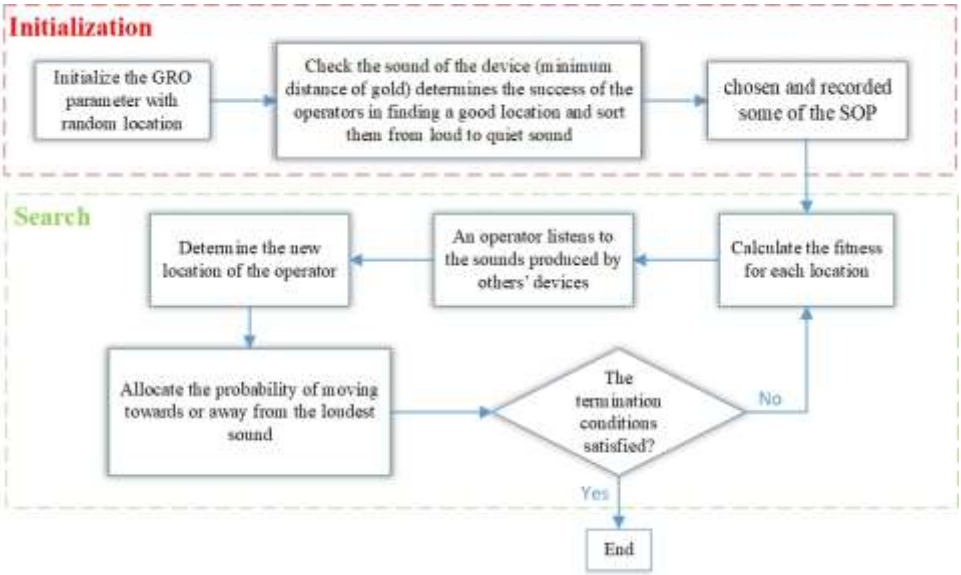

Figure 3. The flowchart of the GRO algorithm

3. RESULTS AND DISCUSSION

The GRO algorithm was benchmarked on several benchmark functions. The benchmark functions by CEC 2005 used by researchers Kamalinejad [44]. These functions are chosen to compare the results with the previous meta-heuristics. 25,72,200,272,582-bar truss problem and five constrained classical engineering design problems, namely cantilever beam, tension/compression spring, welded beam, pressure vessel, three-bar truss, are employed to verify the ability of the proposed GRO algorithm (problems have several constraints). For verifying the results, the GRO algorithm is compared with (QEA) [59], (MVO) [60], (WOA) [44], (CSA) [61], etc. MATLAB version R2018b was used for programming and optimization.

3.1 Benchmark function

A previous study Tsoulos [44] has used several unimodal and multimodal benchmark functions by CEC 2005 and obtained the optimal answer. It is possible to split the benchmark functions used into two groups: unimodal, multimodal. For benchmarking the exploitation of algorithms, used Unimodal benchmark functions that have one global optimum and for benchmarking the exploration of algorithms, used multi-modal benchmark functions that have a global optimum as well as multiple local optima. It should be noted here that the variables numbers of the test functions are also considered to be 30. Using GRO, those functions were re-evaluated, as listed in Table 1. The maximum run was tuned at 20. The coefficients $\alpha = 0.5, \beta = 0.5, \text{and } \gamma = 0.85$ are considered for evaluating benchmark functions. Table 2 summarizes and compared the results of WOA [44], PSO [37], GSA [19], DE [8] and current work (GRO). The performance of the algorithm has been suitable and competitive in unimodal functions (exploitation capacity algorithm). The GRO algorithm can hence provide very good exploitation. In multimodal functions (exploration capacity algorithm) the ability to get the best answer or close to the best answer. A perspective view for some of functions are shown in Fig. 4. result Fig. 5 show that the behavior of the GRO algorithm in most cases is the rapid convergence of the initial iteration steps. Also, a good combination and balance of exploitation and exploitation supports the GRO algorithm, which helps this algorithm find the global optimum.

Table 1: Benchmark functions re-evaluated using GRO (range is the boundary of the function's search space, and f_{min} is optimum)

Unimodal Benchmark function	**V-no**	**variable range**	**f_{min}**
$F_1(x) = \sum_{i=1}^{n} x_i^2$	30	$x \in [-100, 100]$	0.0
$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	$x \in [-10, 10]$	0.0
$F_3(x) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j \right)^2$	30	$x \in [-100, 100]$	0.0
$F_4(x) = \sum_{i=1}^{n} \left[100 \left(x_{i+1} - x_i^2 \right)^2 + \left(x_i - 1 \right)^2 \right]$	30	$x \in [-100, 100]$	0.0

$F_5(x)=\sum_{i=1}^{n} ix_i^4+random[0,1)$	30	$x\in[-1.28,1.28]$	0.0
Multimodal Benchmark function			
$F_6(x)=\sum_{i=1}^{n}-x_i\sin\left(\sqrt{\lvert x_i\rvert}\right)$	30	$x\in[-500,500]$	−418.9829*5
$F_7(x)=\sum_{i=1}^{n}\left[x_i^2-10\cos(2\pi x_i)+10\right]$	30	$x\in[-5.12,5.12]$	0.0
$F_8(x)=-20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right)-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right)+20+e$	30	$x\in[-32,32]$	0.0
$F_9(x)=\frac{1}{4000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1$	30	$x\in[-600,600]$	0.0
$F_{10}(x)=\frac{\pi}{n}\{10\sin(\pi y_1)+\sum_{i=1}^{n-1}(y_i-1)^2\left[1+10\sin^2(\pi y_{i+1})\right]+(y_n-1)^2\}+\sum_{i=1}^{n}u(x_i,10,100,4)$	30	$x\in[-50,50]$	0.0
$y_i=1+\frac{x_i+1}{4}\quad u(x_i,a,k,m)=\begin{cases}k(x_i-a)^m & x_i\succ a\\ 0 & -a\prec x_i\prec a\\ k(-x_i-a)^m & x_i\prec -a\end{cases}$			

Table 2: Comparison of optimization results obtained for the unimodal and multimodal benchmark functions

Function	GRO		WOA		PSO		GSA		DE	
	ave	std	ave	std	ave	std	ave	std	ave	std
F1	4.65E-26	3.20E-09	1 .41E −30	4 .91E −30	0 .0 0 0136	0 .0 0 0202	2 .53E −16	9 .67E −17	8 .2E −14	5 .9E −14
F2	7.34E-24	1.84E-11	1 .06E −21	2 .39E −21	0 .042144	0 .045421	0 .055655	0 .194074	1 .5E −09	9 .9E −10
F3	1.12E-3	3.156	5 .39E −07	2 .93E −06	70 .12562	22 .11924	896 .5347	318 .9559	6 .8E −11	7 .4E −11
F4	19.5756	2.3657	27 .86558	0 .763626	96 .71832	60 .11559	67 .54309	62 .22534	0	0
F5	0.01954	0.0176	0 .001425	0 .001149	0 .122854	0 .044957	0 .089441	0 .04339	0 .00463	0 .0012
F6	-12°ך3	347.9934	−5080.76	695.7968	−4841.29	1152.814	−2821.07	493.0375	−11080.1	574.7
F7	0.0045	4.5687	0	0	46 .70423	11 .62938	25 .96841	7 .470068	69 .2	38 .8
F8	0.05764	1.2E-04	7 .4043	9 .897572	0 .276015	0 .50901	0 .062087	0 .23628	9 .7E −08	4 .2E −08
F9	0.000198	1.2578	0 .0 0 0289	0 .001586	0 .009215	0 .007724	27 .70154	5 .040343	0	0
F10	0.09624	0.4567	0 .339676	0 .214864	0 .006917	0 .026301	1 .799617	0 .95114	7 .9E −15	8E −15

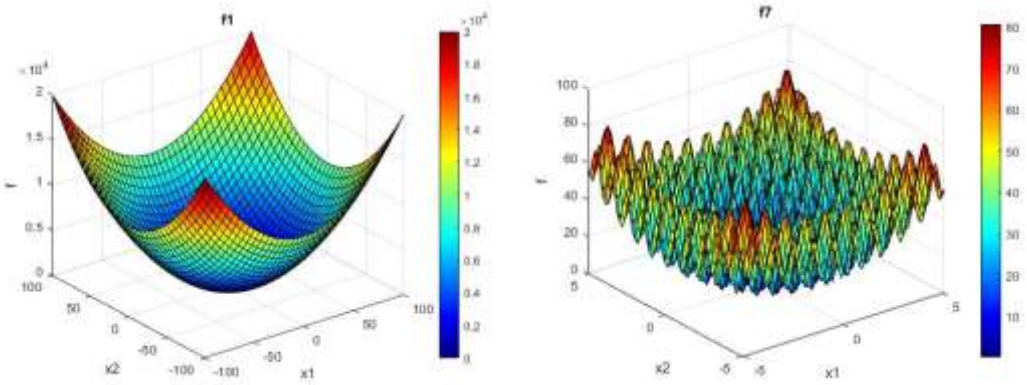

Figure 4. A perspective view for some of the functions

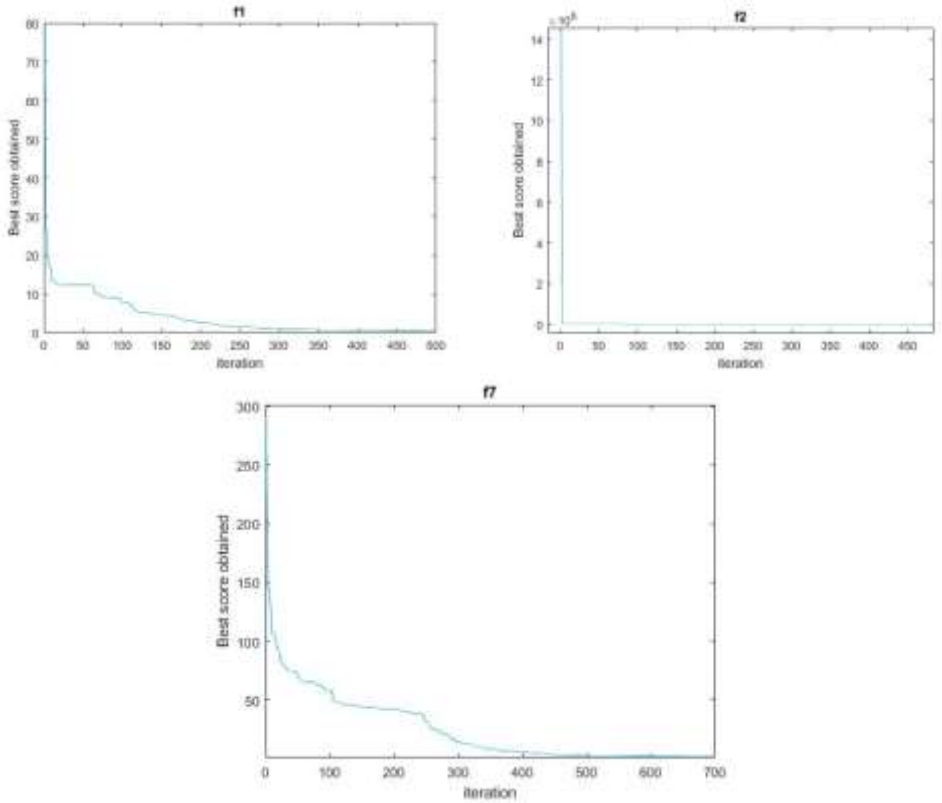

Figure 5. Convergence curve for f1,f2 and f7 functions

3.2 Cantilever beam design problem

Chickermane [62] and Gandomi [63] designed a cantilever beam design problem that includes five hollow elements with a square-shaped cross-section problem, as illustrated in Fig.6. This problem aims to minimize beam weight in which, considering constraints such as vertical constraint displacement of the lowest weight of the beam is taken into account. The

coefficients $\alpha = 0.5, \beta = 0.5, \text{and } \gamma = 0.85$ are considered. The number of population was tuned 20 operators. There are five squares of which the first block is fixed, and the fifth one burdens a vertical load. Fig.6 shows that each element is defined by one variable while the thickness is constant, so there is a total of five structural parameters that define the shape of a cross-section of the cubes. The total cost function, constraints and considered variable ranges are presented in Ref [63].

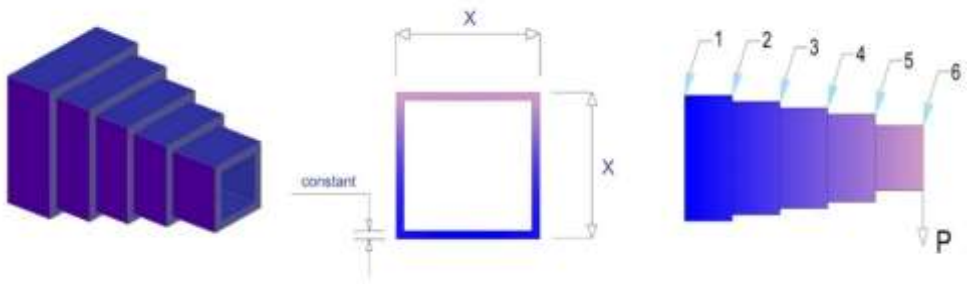

Figure 6. Cantilever beam design problem

Table 3 summarizes the variables and cost functions in the literature and current work. Taking into account the studies performed previously and the several algorithms used to reduce the cost function (cantilever beam), the results obtained for reducing the cantilever beam weight is significantly improved compared to the previous studies (CS) [63], (SOS) [64], (MVO) [60], (QEA) [59]. In previous studies, the lowest cost function (QEA) [59] was 1.3382, whereas this study achieves a cost function of 1.336562, which is considerably lower than previous studies. GRO found minimum weight and constraints are $g(x) = 0.42315$. Based on the results obtained for the cost function in the earlier studies, the constraints of the current research are closer to the boundary conditions. This table shows the optimum values of the decision variables over 30 independent runs corresponding to GRO Best solution. To find a design with the optimum weight by the GRO algorithm, which requires 3300 function evaluations. Fig. 7 illustrates the performance of the GRO concerning the fitness and convergence problem of the cantilever beam design. As can be seen from the convergence curve, the initial slope is suitable and the algorithm reaches the range of good answers in its initial iterations. In fact, the convergence speed GRO algorithm is very good and competitive.

The statistical simulation results (average and standard deviation) are reported in Table 3. As can be seen in Table 3, the standard deviation of the results by GRO is minimal. Furthermore, from Table 3, it can be seen that the worst searching quality of GRO is similar to optimal weight of CS [63], SOS [64], and MVO [60] algorithm. Based on the results of standard deviations, it proves the stability of the algorithm is strong enough. To put it another way, it is an adequate search capability, it is a sufficient search capacity, and convergence is quite good.

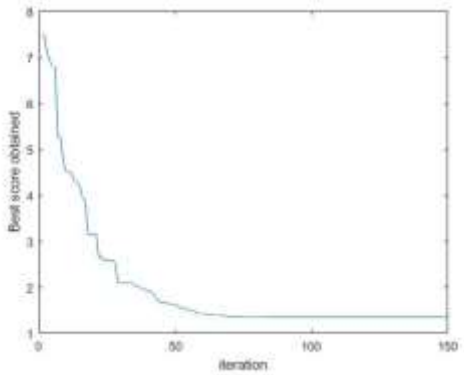

Figure 7. Convergence curve of cantilever beam design problem by the GRO method

Table 3: Comparison of the results for the cantilever beam design problem

Method	Optimal sections (in^2)					Optimal weight	Mean	Worst	Standard deviation
	X_1	X_2	X_3	X_4	X_5				
(**CS**) [63]	6.0089	5.3049	4.5023	3.5077	2.1504	1.33999	N/A	N/A	N/A
(**SOS**) [64]	6.01878	5.30344	4.49587	3.49896	2.15564	1.33996	N/A	N/A	N/A
(**MVO**) [60]	6.02394	5.30601	4.49501	3.4960	2.15272	1.33995	N/A	N/A	N/A
(**QEA**)[59]	5.8594	5.3125	4.6563	3.46875	2.2031	1.3382	1.3532	1.4072	0.015096
Present work (GRO)	6.028099	5.3036577	4.4678013	3.524043	2.1507254	1.336562	1.338997	1.33993	0.001021

(Not available= N/A)

3.3 A tension/compression spring design problem

Belegundu [65] and Arora [66] designed a tension/compression spring problem, as illustrated in Fig.8. This problem aims to minimum deflection, surge frequency, and shear stress of the lowest weight of spring is taken into account. As seen in Fig.8, the number of active coils ($N=X_3$), coil diameter ($D=X_1$), and wire diameter ($d=X_2$) are the considered variables. The coefficients $\alpha = 0.5, \beta = 0.5, \text{and } \gamma = 0.85$ are considered. The number of population was tuned 25 operators. The total cost function, constraints and considered variable ranges are presented in Ref [66].

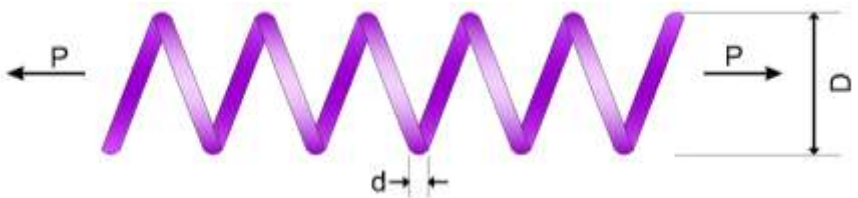

Figure 8. Tension/compression spring design problem

Table 4 summarizes the variables and cost functions in the literature and current work. The results obtained for reducing the (tension/compression spring) weight is significantly improved compared to the previous studies (WOA) [44], (CSA) [61], (MBA) [67], (MCSS) [68]. However, the values found in (MBA) [67] and (CSA) [61] are nearly identical to those

found in GRO. In this study achieves a cost function of 0.01260682. GRO found minimum weight and constraints are $g_1(x) = -5.7422e-07,\ \ g_2(x) = 0.0,\ \ g_3(x) = 4.06593,\ \ g_4(x) = -4.0659$. Based on the results obtained for the cost function in the earlier studies, the constraints of the current research are closer to the boundary conditions. This table shows the optimum values of the decision variables over 30 independent runs corresponding to GRO Best solution. To find a design with the optimum cost function by the GRO algorithm, which requires 4200 function evaluations. Fig. 9 illustrates the performance of the GRO concerning the fitness and convergence problem of the tension/compression spring. As can be seen from the convergence curve, the initial slope is suitable and the algorithm reaches the range of good answers in its initial iterations. In fact, the convergence speed of GRO algorithm is very good and competitive.

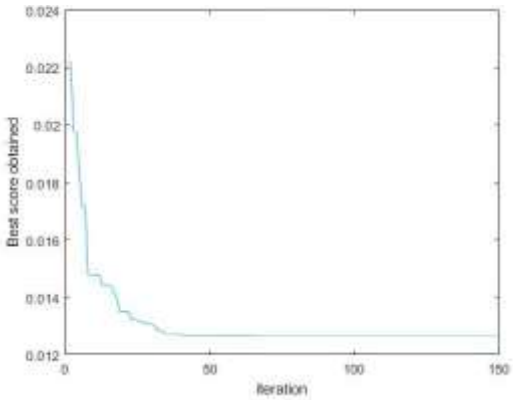

Figure 9. Convergence curve of tension/compression spring design problem by the GRO method

Table 4: Optimum and statistical results for the tension/compression spring design

Methods	Optimal design variables			f_{cost}	Mean	Worst	Standard deviation
	$x_1(d)$	$x_2(D)$	$x_3(N)$				
(WOA) [44]	0 .051207	0 .345215	12 .004032	0 .0126763	0 .0136	N/A	0 .0026
(CSA) [61]	0.051689028	0.356716954	11.28901179	0.01266523	0.012665998	0.0126701	1.357079e_6
(MBA) [67]	0.051656	0.355940	11.344665	0.012665	0.012713	0.012900	6.30e − 05
(MCSS) [68]	0.051627	0.356290	11.275456	0.0126069	0.012712	0.012982	4.7831e − 5
Present work	0.051627	0.356289	11.27546	0.01260682	0.01266991	0.012891	3.86159e-05

(Not available= N/A)

The statistical simulation results (average and standard deviation) are reported in Table 4. Furthermore, from Table 4, it can be seen that the average searching quality of GRO is also better than (WOA) [44], (MBA) [67] and (MCSS) [68]. Based on the results of standard deviations, it proves the stability of the algorithm is strong enough. To put it another way, it is an adequate search capability, it is a sufficient search capacity, and convergence is quite good.

3.4 A welded beam design problem

Rao [69] designed a welded beam design problem, as illustrated in Fig.10. This problem aims to minimize the weight of the welded beam in which, considering constraints such as minimum shear stress (τ), bending stress (σ), buckling load (PC), end deflection (δ). The coefficients $\alpha = 0.75, \beta = 0.6,$ and $\gamma = 0.85$ are considered. The number of population was tuned 25 operators. Fig.10 shows that, the four design variables are: (h= X_1), (l= X_2), (t = X_3), and (b= X_4).

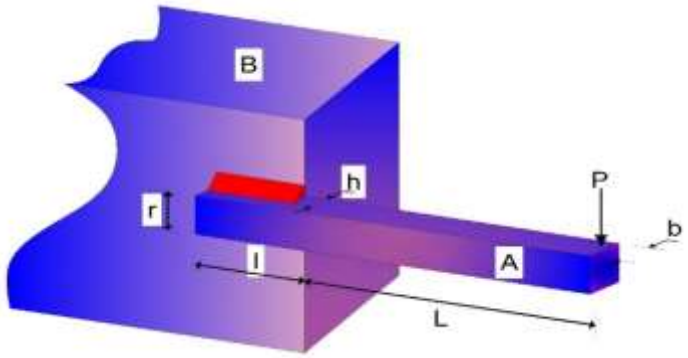

Figure 10. Welded beam design problem

The total cost function, constraints and considered variable ranges are presented in Ref [69].

Table 5 summarizes the variables and cost functions in the literature and current work. Taking into account the studies performed previously and the several algorithms used to reduce the cost function (welded beam), the results obtained for reducing the welded beam weight is significantly improved compared to the previous studies (QEA) [59], (CSA) [61], (WOA) [44] and (SSOA) [70]. In previous studies, the lowest cost function (CSA) [61] was 1. 7248523, whereas this study achieves a cost function of 1.7278560, which is the results are quite similar. GRO found minimum weight and constraints are $g_1(x) = -5.9690,\ g_2(x) = -75.2661,\ g_3(x) = -0.0011,\ g_4(x) = -3.3874,\ g_5(x) = -0.0796,\ g_6(x) = -0.2356,\ g_7(x) = -2.8045$. Based on the results obtained for the cost function in the earlier studies, the constraints of the current research are closer to the boundary conditions. This table shows the optimum values of the decision variables over 30 independent runs corresponding to GRO Best solution. To find a design with the optimum cost function by the GRO algorithm, which requires 40000 function evaluations. Fig. 11 illustrates the performance of the GRO concerning the fitness and convergence problem of the welded beam design. As can be seen from the convergence curve, the initial slope is suitable and the algorithm reaches the range of good answers in its initial iterations. In fact, the convergence speed of GRO algorithm is very good and competitive.

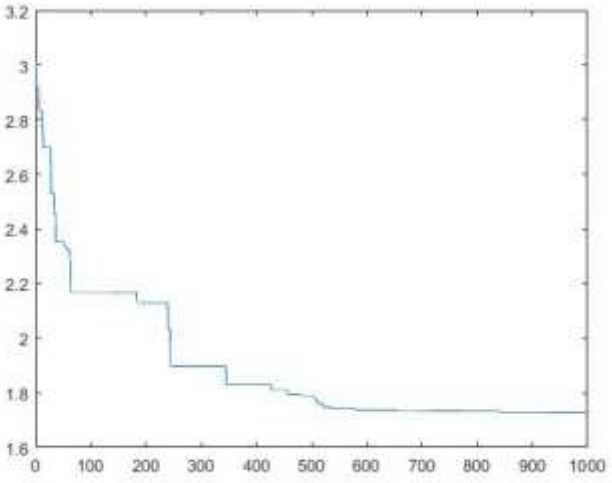

Figure 11. Convergence curve of welded beam design problem by the GRO method

Table 5: Optimum and statistical results for the welded beam design

Methods	Optimal design variables				f_{cost}	Mean	Worst	Standard deviation
	$x_1(h)$	$x_2(l)$	$x_3(t)$	$x_4(b)$				
(QEA)[59]	0.21875	3.1094	8.7656	0.21875	1.742706	1.926865	2.709145	0.197686
(WOA) [44]	0 .205396	3 .484293	9 .037426	0 .206276	1 .730499	1 .7320	N/A	0 .0226
(SSOA) [70]	0.2057296	3.4704888	9.0366236	0.2057297	1.7248524	1.724855	1.724871	4.32E-06
(CSA) [61]	0.2057296	3.4704886	9.0366239	0.2057296	1.7248523	1.724852	1.724852	1.1945e_15
Present work	0.2046242	3.4917238	9.0485875	0.2057020	1.7278560	1.733021	1.740871	0.003094

(Not available= N/A)

The statistical simulation results (average and standard deviation) are reported in Table 5. From Table 5, it can be seen that the worst searching quality of GRO is also better than the QEA algorithms. Based on the results of standard deviations, it proves the stability of the algorithm is strong enough. To put it another way, it is an adequate search capability, it is a sufficient search capacity, and convergence is quite good.

3.5 A pressure vessel design problem

Sandgren [71] designed a pressure vessel design problem that both ends of the vessel are capped, and the head has a hemispherical shape, as illustrated in Fig.12. This problem aims to minimize the total cost consisting of material, forming, and welding of a cylindrical vessel, as shown in Fig.12. The coefficients $\alpha = 0.75, \beta = 0.6, \text{and } \gamma = 0.85$ are considered. The number of population was tuned 20 operators. Fig.12 shows the considered variables are the thickness of the shell (TS=X_1), the thickness of the head (Th= X_2), the inner radius (R= X_3), and the length of the cylindrical section of the vessel (L=X_4), not including the head. TS and Th are integer multiples of 0.0625 inch, R and L, the available thickness of the rolled steel plates, are continuous.

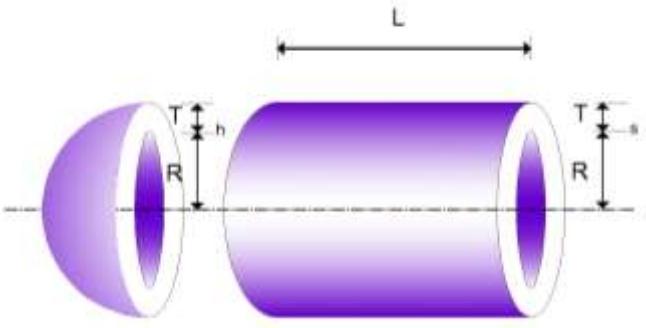

Figure 12. Schematic of a section of pressure vessel in pressure vessel design problem

The total cost function, constraints and considered variable ranges are presented in Ref [71].

Table 6 Optimum results for the pressure vessel

Methods	Optimal design variables				f_{cost}	Mean	Worst	Standard deviation
	$x_1(T_s)$	$x_2(T_h)$	$x_3(R)$	$x_4(L)$				
(MVO) [60]	0.812500	0.437500	42.0907382	176.738690	6,060.8066	N/A	N/A	N/A
(WOA) [44]	0 .812500	0 .43750	42 .098269	176 .63899	6059 .7410	6068 .05	N/A	65 .6519
(CSA) [61]	0.812500	0.437500	42.0984453	176.636598	6059.71436	6342.4991	7332.841	384.945
(MBA) [67]	0.7802	0.3856	40.4292	198.4964	5,889.3216	6200.64765	6392.5062	160.34
(QEA) [59]	0.765625	0.378906	39.5625	211.1641	5,887.3969	6527.735	7391.332	422.98
Present work	0.77818755	0.3846585	40.320598	199.9863563	5,885.30256	6,097.1875	6,291.4618	87.1246

(Not available= N/A)

Table 6 summarizes the variables and cost functions in the literature and current work. Taking into account the studies performed previously and the several algorithms used to reduce the cost function (pressure vessel), the results obtained for reducing the pressure vessel cost function is significantly improved compared to the previous studies (QEA) [59], (MBA) [67], (CSA) [61], (WOA) [44], (MVO) [60]. In previous studies, the lowest cost function (QEA) [59] was 5,887.3969, whereas this study achieves a cost function of 5,885.30256, which is considerably lower than previous studies. GRO found minimum function and constraints are $g_1(x) = -7.5238e-12$, $g_2(x) = -1.3725e-12$, $g_3(x) = 0.0011$, $g_4(x) = -40.0136$. Based on the results obtained for the cost function in the earlier studies, the constraints of the current research are closer to the boundary conditions. This table shows the optimum values of the decision variables over 30 independent runs corresponding to GRO Best solution. To find a design with the optimum cost function by the GRO algorithm, which requires 3300 function evaluations. Fig. 13 illustrates the performance of the GRO concerning the fitness and convergence problem of the welded beam design. As can be seen from the convergence curve, the initial slope is suitable and the algorithm reaches the range of good answers in its initial iterations. In fact, the convergence speed of GRO algorithm is very good and competitive.

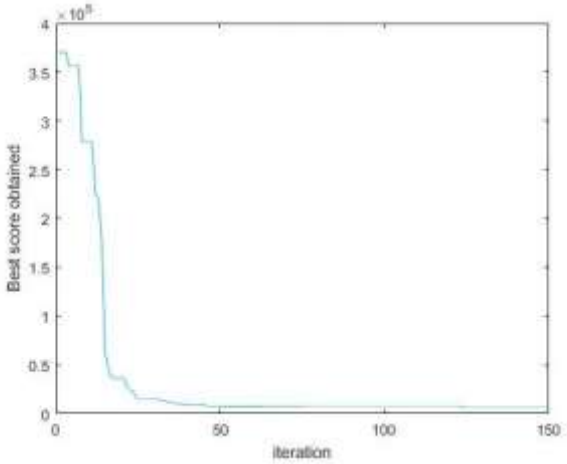

Figure 13. Convergence curve of pressure vessel design problem by the GRO method

The statistical simulation results (average and standard deviation) are reported in Table 6. Based on the results of standard deviations, it proves the stability of the algorithm is strong enough. To put it another way, it is an adequate search capability, it is a sufficient search capacity, and convergence is quite good.

3.6 The three-bar truss design problem

Sadollah [67] and Gandomi [63] designed a three-bar truss design problem, as illustrated in Fig.14. This problem aims to minimize the weight of the truss. The coefficients $\alpha = 0.75, \beta = 0.6, \text{and } \gamma = 0.85$ are considered. The number of population was tuned 20 operators. The considered constraint satisfying the stress, deflection, and buckling is taken into account. The considered variables are the cross-section of the truss bar. Fig.13 shows A_1, A_2, and A_3 are a cross-section of the truss bar also $A_1 = A_3$ are considered.

The total cost function, constraints and considered variable ranges are presented in Ref [67].

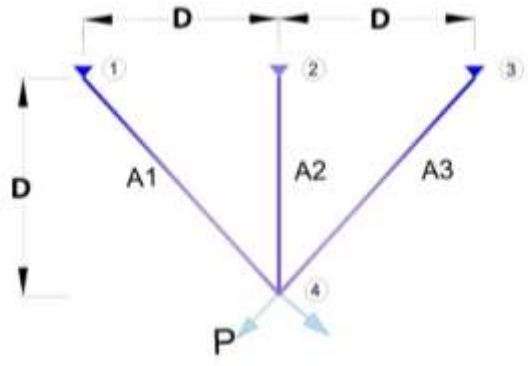

Figure 14. The three-bar truss design problem (A1=A3)

Table 7 summarizes the variables and cost functions in the literature and current work. The

studies performed previously and the several algorithms used to reduce the cost function (three-bar truss). The results obtained is significantly improved compared to the previous studies (MVO) [60], (MBA) [67], (CSA) [61], (CS) [63], (QEA) [59]. In previous studies, the lowest cost function (CSA) [61] was 263.8958433, whereas this study achieves a cost function of 263.895843, which is the results are quite similar. GRO found minimum weight and constraints are $g_1(x) = -8.72857e-13,\ g_2(x) = 1.46412,\ g_3(x) = 0.535876$. Based on the results obtained for the cost function in the earlier studies, the constraints of the current research are closer to the boundary conditions. This table shows the optimum values of the decision variables over 30 independent runs corresponding to GRO Best solution. To find a design with the optimum weight by the GRO algorithm, which requires 3300 function evaluations. Fig. 15 illustrates the performance of the GRO concerning the fitness and convergence problem of the cantilever beam design. As can be seen from the convergence curve, the initial slope is suitable and the algorithm reaches the range of good answers in its initial iterations. In fact, the convergence speed of GRO algorithm is very good and competitive.

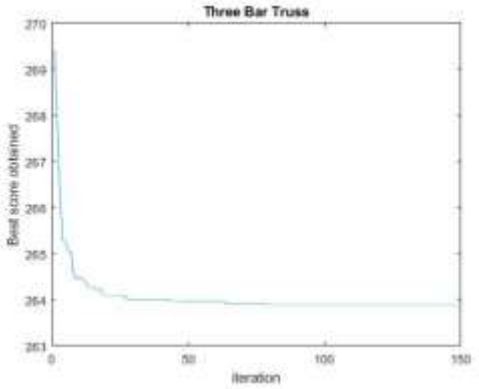

Figure 15. Convergence curve of three-bar truss design problem by the |GRO method

Table 7: Comparison of the results for the three-bar truss design problem

Method	Optimal sections (in^2)		Optimal truss weight	Mean	Worst	Standard deviation
	X_1	X_2				
(**CS**) [63]	0.78867	0.40902	263.9716	264.0669	N/A	0.00009
(**QEA**) [59]	0.792969	0.396484	263.9339	N/A	N/A	N/A
(**MBA**) [67]	0.7885650	0.4085597	263.8958522	263.897996	263.915983	3.93e_3
(**MVO**) [60]	0.78860276	0.40845307	263.8958499	N/A	N/A	N/A
(**CSA**) [61]	0.7886751284	0.408248308	263.8958433765	263.8958	263.8958	1.012254e-10
Present work (GRO)	0.7886819160	0.408229110	263.89584341	263.89476	263.89594	7.13842e-3

(Not available= N/A)

The statistical simulation results (average and standard deviation) are reported in Table 7. Based on the results of standard deviations, it proves the stability of the algorithm is not strong enough. To put it another way, it is an adequate search capability, it is a sufficient search capacity, but convergence is not quite right.

3.7 A 25-bar spatial truss structure

Kaveh [45] and Hasançebi [72] studied a 25-bar spatial truss design problem, which is a widespread problem in the previous study. The schematic topology and element numbering, as illustrated in Fig. 16. there are 10 nodes, of which four are fixed.

The elements (cross-sectional area members) are classified into 8 design groups. All of the groups are presented in [45]. Therefore, this problem has 8 parameters. The material properties assumptions for this problem are as follows: Material density (ρ)= 0.0272 N / cm3 (0.1 lb/in3), Modulus of elasticity (E) = 68.947 MPa (10,000 ksi), Displacement limitation = 0.35 in, Maximum displacement = 0.3504 in. Stress limitations of the member for the 25-bar truss design problem and the subject to two loading conditions are presented in [45].

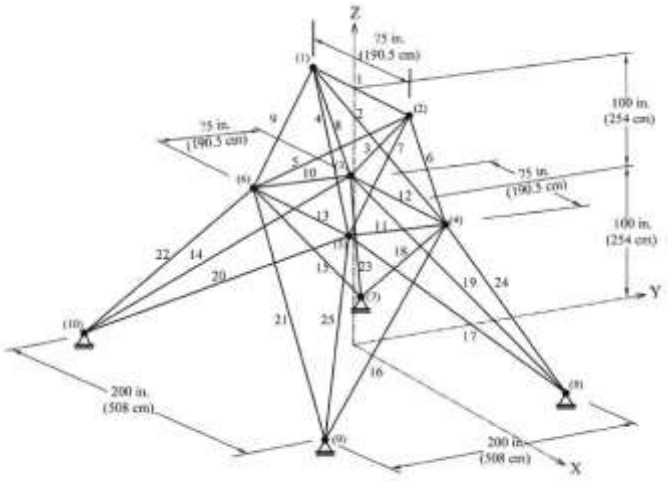

Figure 16. 25-bar spatial truss in design problem (Photo reproduced from [45])

This example has been studied in both continuous and discrete forms. In this study solve the continuous version of this problem. The coefficients α = 0.5, β = 0.5, and γ = 0.85 are considered. The number of population was tuned 20 operators. Table 8 summarizes the cross-sectional area members and shows the best optimal weight in the literature and current work.

Table 8: Performance comparison for the 25-bar truss problem

Methods	Element group (Optimal sections (in2))								Weight (lb)	Mean	Worst	Standard deviation
	1	2	3	4	5	6	7	8				
EBA [73]	0.0100	1.9789	3.0047	0.0100	0.0100	0.6888	1.6783	2.6527	545.1688	546.4464	N/A	N/A
TLBO [48]	0.0100	2.0712	2.9570	0.0100	0.0100	0.6891	1.6209	2.6768	545.0900	545.41	N/A	0.42
WEO [33]	0.0100	1.9814	3.0023	0.0100	0.0100	0.6827	1.6778	2.6612	545.166	N/A	N/A	N/A
GWO [13]	0.0159	1.8017	3.4000	0.0399	0.0164	0.6334	1.7062	2.6149	549.3771	564.1920	N/A	6.9961
IGWO [13]	0.0124	1.9624	3.0204	0.0266	0.0109	0.6841	1.6862	2.6526	545.4819	549.6747	N/A	2.8113
Present work	0.0100	1.9347	3.0517	0.0100	0.0101	0.6956	1.6953	2.6253	545.1519	546.0214	546.8697	0.55183

(Not available= N/A)

Taking into account the results obtained for reducing the 25-bar truss weight is significantly improved compared to the previous studies 545.1688, 545.166, 549.3771 and 545.4819 lb for the EBA [73], WEO [33], GWO [13] and IGWO [13] algorithm, respectively. In previous studies, the lowest welded beam weight TLBO [48] was 545.0900. whereas this study achieves a weight of 545.1519, which is the results are quite similar. To find a design with the optimum weight by the GRO algorithm, which requires 11000 function evaluations. Fig. 17 shows the convergence diagrams in terms of the number of iterations for this example. The maximum values of displacements in the x, y, and z-directions are 8 (in), 7.61 (in), and 2.15 (in), respectively. Based on the results, the variables of the current research are closer to the boundary conditions. The statistical simulation results (average and standard deviation) are lists in Table 8. As can be seen from the convergence curve, the initial slope is suitable and the algorithm reaches the range of good answers in its initial iterations. In fact, the convergence speed of GRO algorithm is very good and competitive.

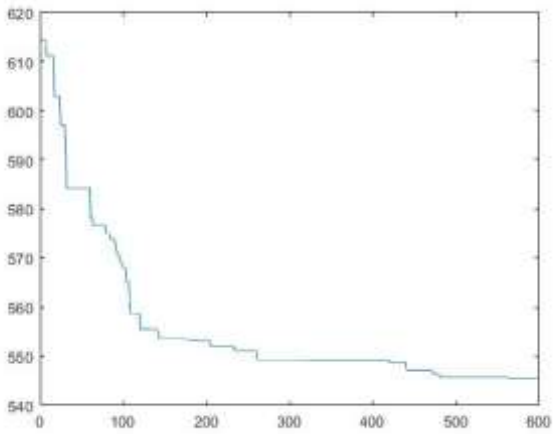

Figure 17. 25-bar spatial skeletal tower design problem

3.8 A 72-bar spatial truss structure

Aslani [74] and Kaveh [75] studied a 72-bar spatial truss design problem, which is a widespread problem in the previous study. The schematic topology and element numbering, as illustrated in Fig. 18. There are 20 nodes, of which four are fixed.

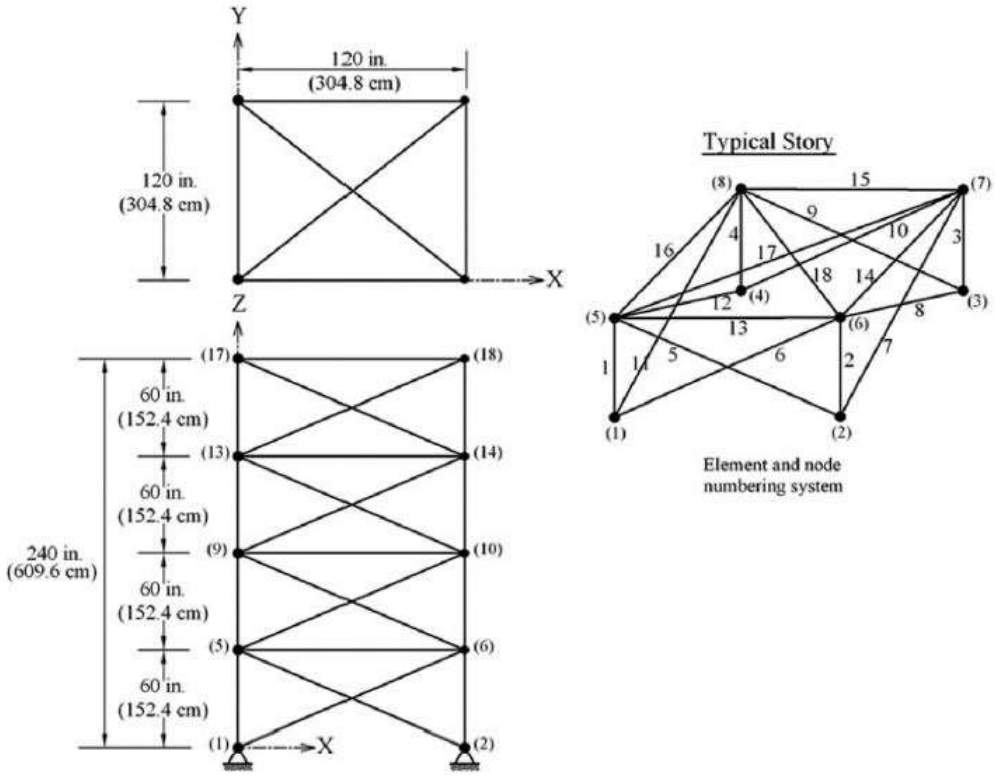

Figure 18. 72-bar spatial truss in design problem (Photo reproduced from [75])

The material properties assumptions for this problem are as follows: Material density (ρ)= 0.1 lb/in^3 (2767.990 kg/m^3), Modulus of elasticity (E) = 10,000 (ksi) (68,950 MPa). The elements (cross-sectional areas members), which are classified into 16 design groups. All of the groups are presented in Ref [74]. Therefore, this problem has 16 parameters.

The minimum and the maximum cross-sectional area of each member is taken as 0.10 in2 (0.6452 cm2) and 4.00 in2 (25.81 cm2). The stress limitations of members for the 72-bar truss design problem are limits of ±25 ksi (±172.375 MPa). Displacement limits of uppermost nodes are subjected to the ±0.25 in (±0.635 cm) in x and y directions. The loading conditions are presented in Ref [74].

This example has been studied in both continuous and discrete forms. In this study solve the continuous version of this problem. The coefficients α = 0.5, β = 0.5, and γ = 0.85 are considered. Using 30 operators over 200 iterations. Table 9 summarizes the cross-sectional area members and shows the best optimal weight in the literature and current work.

Table 9: Performance comparison for the spatial 72-bar tower truss problem

Element group	Optimal sections					
	EBA [73]	CBO [22]	GWO [13]	IGWO [13]	SSOA [70]	GRO present work
1	1.8592	1.9170	2.0347	1.8585	1.8823	1.8501
2	0.4931	0.5031	0.4546	0.5021	0.5126	0.5128
3	0.1003	0.1000	0.1094	0.1002	0.1003	0.1000
4	0.1018	0.1001	0.1000	0.1000	0.1001	0.1000
5	1.2853	1.2721	1.5148	1.3011	1.2442	1.2450
6	0.5131	0.5050	0.5876	0.5151	0.5161	0.5159
7	0.1007	0.1000	0.1970	0.1000	0.1000	0.1000
8	0.1025	0.1000	0.1501	0.1001	0.1001	0.1000
9	0.5121	0.5184	0.3973	0.5311	0.5312	0.53101
10	0.5255	0.5362	0.4312	0.5122	0.5166	0.51645
11	0.1003	0.1000	0.1000	0.1008	0.1000	0.1000
12	0.1030	0.1000	0.1000	0.1030	0.1006	0.1000
13	0.1560	0.1569	0.2157	0.1560	0.1562	0.1562
14	0.5547	0.5374	0.6051	0.5472	0.5486	0.5486
15	0.4063	0.4062	0.3792	0.4202	0.4027	0.4027
16	0.5962	0.5741	0.8182	0.5793	0.5726	0.5955
Weight (lb)	380.0582	379.75	400.3635	379.7615	379.6699	379.6241
Average optimized weight (lb)	389.1439	380.03	409.9525	380.6811	379.9030	379.9004
Standard deviation on average weight (lb)	N/A	0.278	4.7978	0.7315	0.1150	0.1173

Taking into account the results obtained for reducing the 72-bar truss weight is better than previous algorithms. Results show that the best optimal weight obtained by GRO is 379.6241 while it is 380.0582, 379.75, 400.3635, 379.7615 and 379.6699 lb for the EBA [73], CBO [22], GWO [13], IGWO [13] and SSOA [70] algorithm, respectively. To find a design with the optimum weight by the GRO algorithm, which requires 33000 function evaluations. Fig. 19 shows the convergence diagrams in terms of the number of iterations for this example. As can be seen from the convergence curve, the initial slope is suitable and the algorithm reaches the range of good answers in its initial iterations. In fact, the convergence speed of GRO algorithm is very good and competitive.

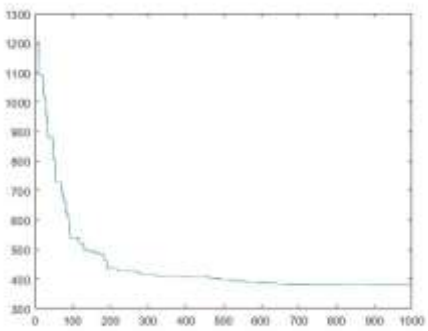

Figure 19. 72-bar spatial skeletal tower design problem

Based on the results, the variables of the current research are closer to the boundary conditions. The statistical simulation results (average and standard deviation) are lists in Table 9. As can be seen, the standard deviation shows that GRO is very small. Furthermore, it can be seen that the average searching quality of GRO is also better than other algorithm.

3.9 A 200-bar planar truss structure

Kaveh and Zaerreza [70] studied a 200-bar planer truss design problem. The schematic topology and element numbering, as illustrated in Fig. 20. There are 77 nodes, of which two are fixed. The members are all made of steel.

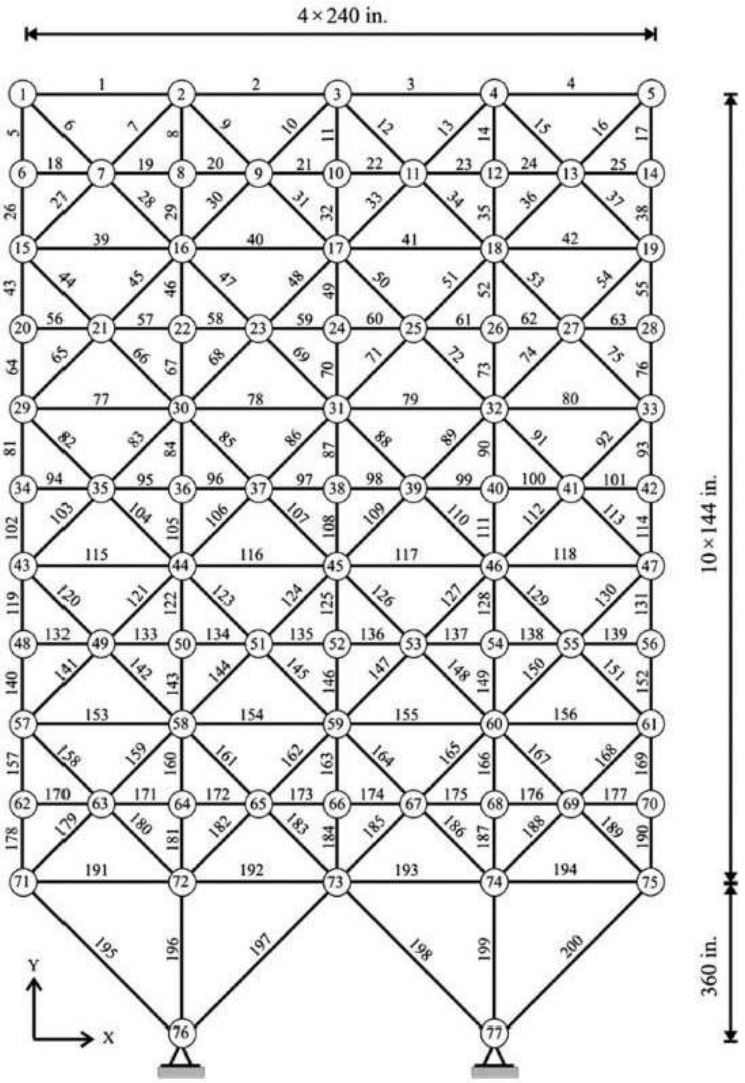

Figure 20. 200-bar spatial truss in design problem(Photo reproduced from[70])

The material properties assumptions for this problem are as follows: Material density (ρ)= 0.283 lb/in^2, Modulus of elasticity (E) = 30,000 (ksi). The elements (cross-sectional areas members), which are classified into 29 design groups. All of the groups are presented in Ref [70]. Therefore, this problem has 29 parameters.

The minimum and the maximum cross-sectional area of each member is taken as 0.10 in^2 and 20 in^2. Stress limitations of members for the 200-bar truss design problem are limits of ±10 ksi. The optimization process does not include any displacement constraints. The loading conditions are presented in Ref [70].

In this study solve the continuous version of this problem. The coefficients α = 0.5, β = 0.5, and γ = 0.85 are considered. Using 30 operators over 200 iterations. Table 10 summarizes the cross-sectional area members and shows the best optimal weight in the literature and current work.

Table 10: Performance comparison for the planar 200-bar tower truss problem

Element group	Optimal sections				
	TLBO [48]	WEO [33]	CSP [75]	SSOA [70]	GRO present work
1	0.1460	0.1144	0.1480	0.100315	0.145800
2	0.9410	0.9443	0.9460	0.945976	0.940145
3	0.1000	0.1310	0.1010	0.101669	0.100000
4	0.1010	0.1016	0.1010	0.101119	0.100023
5	1.9410	2.0353	1.9461	1.948657	1.940580
6	0.2960	0.3126	0.2979	0.290019	0.295755
7	0.1000	0.1679	0.1010	0.101494	0.100000
8	3.1210	3.1541	3.1072	3.113037	3.119168
9	0.1000	0.1003	0.1010	0.100209	0.100000
10	4.1730	4.1005	4.1062	4.118997	4.170924
11	0.4010	0.4350	0.4049	0.407309	0.401063
12	0.1810	0.1148	0.1944	0.100247	0.181012
13	5.4230	5.3823	5.4299	5.406352	5.422000
14	0.1000	0.1607	0.1010	0.109823	0.100067
15	6.4220	6.4152	6.4299	6.406481	6.422000
16	0.5710	0.5629	0.5755	0.470731	0.570145
17	0.1560	0.4010	0.1349	0.430735	0.156000
18	7.9580	7.9735	7.9747	7.968047	7.957485
19	0.1000	0.1092	0.1010	0.119819	0.100000
20	8.9580	9.0155	8.9747	8.974186	8.957422
21	0.7200	0.8628	0.70648	0.888874	0.720022
22	0.4780	0.2220	0.4225	0.226645	0.477602
23	10.8970	11.0254	10.8685	11.14610	10.897000
24	0.1000	0.1397	0.1010	0.221872	0.100080
25	11.8970	12.0340	11.8684	12.14581	11.897000
26	1.0800	1.0043	1.0340	1.096515	1.079267
27	6.4620	6.5762	6.6859	5.72775	6.460494
28	10.7990	10.7265	10.8111	10.35575	10.798389
29	13.9220	13.9666	13.8465	14.18211	13.921755
Weight (lb)	25488.15	25674.83	25476.9	25291.024	25484.0022
Max (σ_{member})	9.99997	9.99696	10.01273	11.44526	10.00000
Average optimized	25533.14	26613.45	25547.6	25763.978	25694.943

weight (lb)					
Standard deviation on average weight (lb)	27.44	702.80	135.09	270.444	298.481

Taking into account the results obtained for reducing the 200-bar truss weight is closer to previous algorithms. Results show that the best optimal weight obtained by GRO is 25484.0022 while it is 25488.15, 25674.83, 25476.9 and 25291.024 lb for the TLBO [48], WEO [33], CSP [75] and SSOA [70] algorithm, respectively. To find a design with the optimum weight by the GRO algorithm, which requires 23000 function evaluations. Fig. 21 shows the convergence diagrams in terms of the number of iterations for this example. As can be seen from the convergence curve, the initial slope is suitable and the algorithm reaches the range of good answers in its initial iterations. In fact, the convergence speed of GRO algorithm is very good and competitive.

Based on the results, the variables of the current research are closer to the boundary conditions. The statistical simulation results (average and standard deviation) are lists in Table 10. Furthermore, based on the results of the standard deviation, it shows that the stability of the algorithm is strong enough. As was shown in Table.10, some methods with better answers violated the stress constraint.

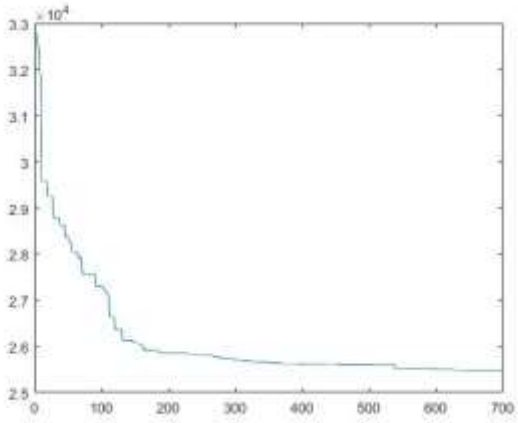

Figure 21. 200-bar spatial skeletal tower design problem

3.10. A 272-bar transmission truss structure

Kaveh and Massoudi [76] studied a 272-bar planer truss design problem. The schematic topology and element numbering, as illustrated in Fig. 22. There are 65 nodes, of which four are fixed. The members are all made of steel.

The material properties assumptions for this problem are as follows: Modulus of elasticity E = 2e8 (KN/m2). The elements (cross-sectional areas members), which are classified into 28 design groups. All of the groups are presented in Ref [76]. Therefore, this problem has 28 parameters. All nodal coordinate and member end nodes are shown in Ref [76].

The minimum and the maximum cross-sectional area of each member is taken as 0.10 in2 and 20 in2. Stress limitations of members for the 272-bar truss design problem are limits of

±275000 KN/m2. Displacement limits of nodes 1,2,11,20,29 are subjected to the 100 mm in the x and y-directions and 20 mm in the z-direction. The loading conditions are presented in Ref [76].

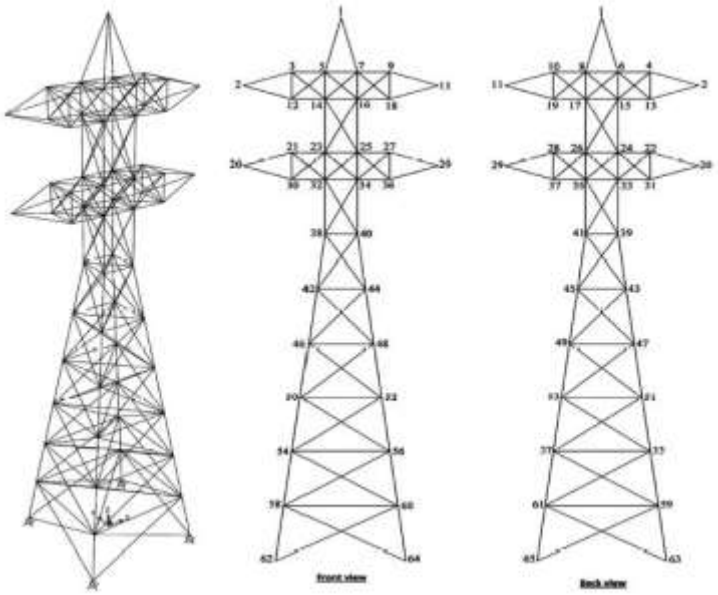

Figure 22. 272-bar spatial truss in design problem(Photo reproduced from[76])

In this study solve the continuous version of this problem. The coefficients α = 0.5, β = 0.5, and γ = 0.85 are considered. Using 30 operators over 200 iterations. Table 11 summarizes the cross-sectional area members and shows the best optimal weight in the literature and current work.

Table 11: Performance comparison for the 272-bar transmission truss structure

Element group	Optimal sections (mm2)		Element group	Optimal sections (mm2)		Element group	Optimal sections (mm2)		Element group	Optimal sections (mm2)	
	SSOA [70]	GRO present work		SSOA [70]	GRO present work		SSOA [70]	GRO present work		SSOA [70]	GRO present work
1	1,000.551	1000.0002	**8**	1,001.777	1001.1048808	**15**	9,320.549	9321.0719492	**22**	1,003.288	1000.0000341
2	1,240.013	1239.9505	**9**	1,000.188	1000.0001337	**16**	1,000.028	1000.0000458	**23**	7,982.259	7982.2199375
3	2,491.871	2491.8674	**10**	1,000.457	1000.0007011	**17**	1,000.307	1000.0042656	**24**	1,000.445	1000.0000537
4	1,017.829	1017.6729	**11**	10,217.022	10217.068875	**18**	1,002.518	1000.0004589	**25**	1,000.591	1000.0000409
5	9,618.809	9618.8241	**12**	1,000.064	1000.0000351	**19**	8,389.809	8389.699625	**26**	1,000.053	1000.0009980
6	1,000.000	1000.0	**13**	1,000.015	1000.0001074	**20**	1,000.814	1000.0000351	**27**	7,504.298	7504.2977558
7	12,063.816	12063.829366	**14**	1,000.005	1000.0001171	**21**	1,000.004	1000.0000937	**28**	1,000.076	1000.0007226
Volume (cm3)		1,168,200.624	1,168,069.32690								
Average volume (cm3)		1,168,668.715	1,168,701.004								
Standard deviation (cm3)		310.7557	339.2781								

Taking into account the results obtained for reducing the 272-bar truss volume is better than the previous algorithm. Results show that the best optimal volume obtained by GRO is 1,168,069.32690, while it is 1,168,200.624 for the SSOA [70] algorithm, respectively. To find a design with the optimum volume by the GRO algorithm, which requires 23000 function evaluations. Fig. 23 shows the convergence diagrams in terms of the number of iterations for this example. As can be seen from the convergence curve, the initial slope is suitable and the algorithm reaches the range of good answers in its initial iterations. In fact, the convergence speed of GRO algorithm is very good and competitive. Based on the results, the variables of the current research are closer to the boundary conditions. The statistical simulation results (average and standard deviation) are lists in Table 11. Furthermore, based on the results of the standard deviation, it shows that the stability of the algorithm is strong enough.

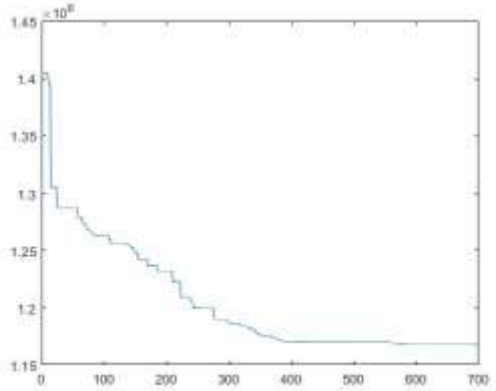

Figure 23. 272-bar spatial skeletal tower design problem

3.11 A 582-bar spatial truss tower

Sonmez [77] studied a 582-bar spatial skeletal tower design problem. The schematic topology and element numbering, as illustrated in Fig. 24. There are 153 nodes, of which 10 are fixed. This problem has been studied with both discrete [72] and continuous [77] variables. In this study, we have been considered the continuous variables.

The material properties assumptions for this problem are as follows: Material density ρ= 0.28 lb/in^3 (7.833 t/m^3), Modulus of elasticity (E) = 29,000 ksi (200 GPa). The elements (cross-sectional areas members), which are classified into 32 design groups according to Fig. 24. Therefore, this problem has 32 parameters. The minimum and the maximum cross-sectional area of each member is taken as 0.99 in^2 (6.45 cm^2) and 49.91 in^2 (322 cm^2). Stress limitations of members for the 582-bar truss design problem are used as specified by the ASD-AISC [78] code, as follows:

$$\begin{cases} \sigma_i^+ = 0.6F_y & for\ \sigma_i \geq 0 \\ \sigma_i^- & for\ \sigma_i \leq 0 \end{cases} \tag{8}$$

Yield stress of steel (F_y) = 124 psi (253.1 MPa). Slenderness ratio dividing the elastic and inelastic buckling regions (C_C) = $\sqrt{2\pi^2 E/F_y}$. Where σ_i^- is calculated according to the slenderness ratio:

$$\sigma_i^- = \begin{cases} \left[\left(1-\frac{\lambda_i^2}{2C_C^2}\right)F_y\right] \Big/ \left[\frac{5}{3}+\frac{3\lambda_i}{8C_C}-\frac{\lambda_i^3}{8C_C^3}\right] & for\ \lambda_i < C \\ \frac{12\pi^2 E}{23\lambda_i^2} & for\ \lambda_i \geq C_C \end{cases} \quad (9)$$

Slenderness ratio $(\lambda_i) = kl_i/r_i$, Effective length factor $(k) = 1$, Member length (l_i), Radius of gyration (r_i) Displacement limits of uppermost nodes are subjected to the 8.0 cm in all directions. The loading conditions are presented in Ref [79].

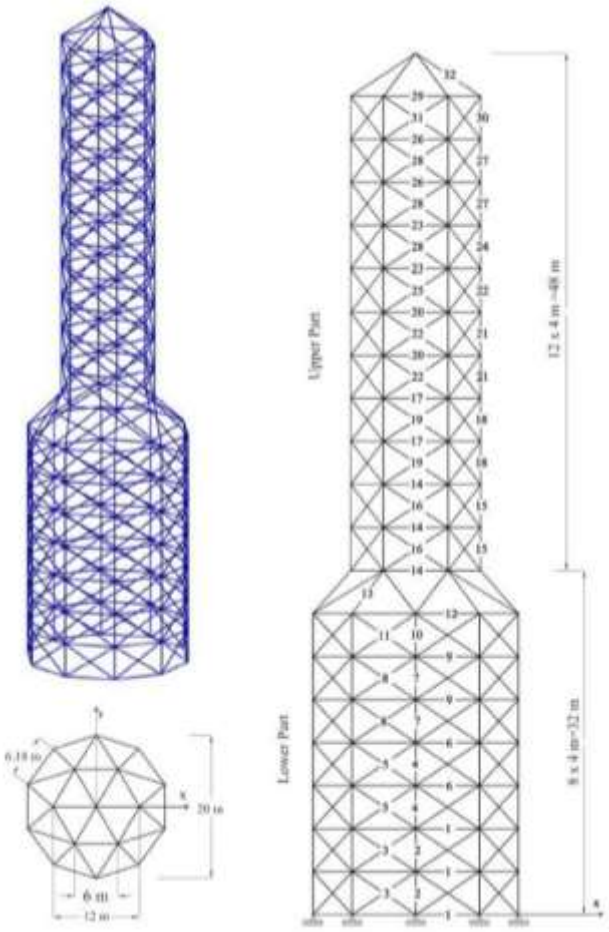

Figure 24. 582-bar spatial skeletal tower design problem (Photo reproduced from (Sonmez

2018))

The coefficients α = 0.7, β = 0.5, and γ = 0.7 are considered. Using 50 operators over 1000 iterations. Table 12 summarizes and lists the optimal values of the cross-sectional area members and shows the best optimal weight in the literature and current work.

Table 12: Performance comparison for the spatial 582-bar tower truss problem

Element group	Optimal sections (in2)					
	ABC	ACO	FA	GWO	Jaya	GRO present work
1	19.35	19.26	19.26	19.26	19.26	8.99
2	168.72	154.35	156.32	148.07	157.87	180.17
3	36.62	36.61	36.57	36.64	36.56	26.95
4	126.89	130.32	121.31	132.60	126.99	143.20
5	33.49	33.38	33.44	33.36	33.39	21.37
6	19.28	19.26	19.29	19.34	19.26	14.04
7	111.31	104.20	100.53	106.00	104.46	125.88
8	30.55	30.51	30.54	30.49	30.50	17.54
9	19.27	19.25	19.25	19.44	19.27	8.65
10	136.07	90.37	96.20	90.47	95.33	100.60
11	27.36	27.65	27.70	27.65	27.61	13.42
12	148.40	141.96	126.50	147.69	141.78	213.92
13	143.60	156.74	168.58	162.03	155.98	168.15
14	93.70	103.11	109.58	104.47	102.76	102.35
15	190.20	168.09	135.30	159.36	166.31	203.78
16	40.48	35.85	37.71	36.39	35.86	44.37
17	18.34	18.37	19.25	18.15	18.14	14.82
18	97.90	132.71	144.45	119.21	131.71	152.13
19	31.13	28.41	28.52	28.61	28.39	26.42
20	18.29	18.31	18.14	18.57	18.15	14.85
21	84.64	87.97	91.18	87.66	90.63	90.75
22	26.93	26.20	26.20	26.30	26.22	37.94
23	18.34	18.19	18.34	20.51	18.21	8.48
24	53.60	52.80	50.45	51.54	50.77	109.17
25	26.29	26.28	26.39	26.29	26.26	20.54
26	18.27	18.24	18.15	18.80	18.15	14.48
27	30.99	21.98	79.85	26.37	22.81	50.34
28	30.99	26.26	26.23	26.21	26.22	13.54
29	18.22	19.07	33.84	21.36	18.17	8.65
30	8.98	9.20	18.53	18.21	8.38	8.89
31	26.29	26.22	28.35	26.34	26.21	13.93
32	26.23	26.31	41.00	28.35	26.25	6.50
Weight (t)	134.993	133.131	136.417	133.709	133.066	129.9826
Average optimized weight (t)	136.744	133.614	142.333	134.305	133.081	130.984
Standard deviation on average weight (t)	0.979	1.618	3.147	0.313	0.010	0.163

All algorithms are from Sonmez [77]

Taking into account the results obtained for reducing the 582-bar truss volume is better than previous algorithms. Results show that the best volume obtained by GRO is 129.9826, while it is 133.131, 134.993, 136.417, 133.709, and 133.066 lb for the ACO, ABC, FA, GWO and Jaya [77] algorithm, respectively. To find a design with the optimum weight by the GRO algorithm, which requires 55000 function evaluations. Fig. 25 shows the convergence diagrams in terms of the number of iterations for this example. As can be seen from the convergence curve, the initial slope is suitable and the algorithm reaches the range of good answers in its initial iterations. In fact, the convergence speed of GRO algorithm is very good and competitive.

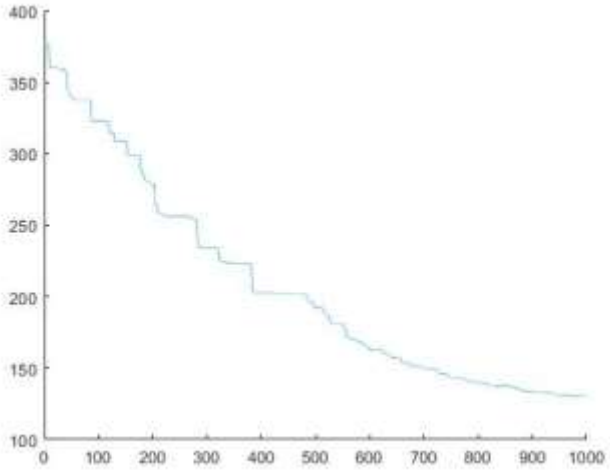

Figure 25. 582-bar spatial skeletal tower design problem

Based on the results, the variables of the current research are closer to the boundary conditions. The statistical simulation results (average and standard deviation) are lists in Table 12. Furthermore, based on the results of the standard deviation, it shows that the stability of the algorithm is strong enough.

5. CONCLUSIONS

In this research, a new approach called GRO was developed based on the thinking and decision-making abilities of a human. The algorithm achieved the ideal CEC 2005 benchmark function using the minimum number of iterations, therefore increasing calculation speed. Compared with previous engineering design examples such as the benchmark function, engineering examples including cantilever beam, tension/compression spring, pressure vessel, three-bar truss and a truss with 72, 272, and 582-bar, the proposed algorithm in this work has been able to achieve better results and a lower cost function, which leads to a reduced structure weight. But in 25-bar, welded beam and 200 bar truss, the proposed algorithm results close to best answer and previous algorithms performed better.

Consequently, this algorithm can be used as an alternative for previous algorithms used to optimize skeletal and continuous structures. The proposed future research should focus on multi-objective GRO and Sense of sight should be considered along with sense of hearing

REFERENCES

1. Holland JH. Adaptation in Natural and Artificial Systems. Ann Arbor : University of Michigan Press; 1975.
2. Goldberg DE. Genetic Algorithms in Search, Optimization and Machine Learning, 1st ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co, Inc, 1989.
3. Ding Y, Zhou K, Bi W. Feature selection based on hybridization of genetic algorithm and competitive swarm optimizer. Soft Computing 2020, doi:10.1007/s00500-019-04628-6.
4. Liang Y, Wang L. Applying genetic algorithm and ant colony optimization algorithm into marine investigation path planning model, Soft Comput 2019; 4. doi:10.1007/s00500-019-04414-4.
5. Kooshkbaghi M, Kaveh A. Sizing Optimization of Truss Structures with Continuous Variables by Artificial Coronary Circulation System Algorithm, Iranian J Sci Technol, Transact Civil Eng 2019; (0123456789). doi:10.1007/s40996-019-00254-2.
6. Koza JR. Genetic Programming:On the Programming of Computers by Means of Natural Selection, (Francisco J, Varela, Paul B, eds). Cambridge, Massachusetts: MIT Press; 1992.
7. Simon D. Biogeography-Based Optimization. IEEE Transact Evolut Computat 2008; 12:702–713. doi:10.1109/TEVC.2008.919004.
8. Storn R, Price K. Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, J Global Optim 1997; 11(4): 341–59. doi:10.1023/A:1008202821328.
9. Yao X, Liu Y. Guangming Lin. Evolutionary programming made faster, IEEE Transact Evolut Computat 1999; 3(2): 82–102. doi:10.1109/4235.771163.
10. Fogel D. Artificial Intelligence through Simulated Evolution, 2009.
11. Hansen N, Müller SD, Koumoutsakos P. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES), Evolut Computat 2003; 11(1): 1–18. doi:10.1162/106365603321828970.
12. Rechenberg I. Evolution Strategy: Comput Intel Imitat Life, Springer, Berlin, Heidelberg; 1994.
13. Kaveh A, Zakian P. Improved GWO algorithm for optimal design of truss structures. Eng Comput 2018; 34(4): 685–707. doi:10.1007/s00366-017-0567-1.
14. Webster B, Bernhard PJ. A local search optimization algorithm based on natural principles of gravitation, In: Proceedings of the 2003 International Conference on Information and Knowledge Engineering (IKE'03), Las Vegas, Nevada, USA; 2003: 255–61.
15. Kaveh A, Ilchi Ghazaan M. A new meta-heuristic algorithm: vibrating particles system, Sci Iranica 2017; 24(2): 551-66. doi:10.24200/sci.2017.2417.
16. Kaveh A, Ilchi Ghazaan M. A new hybrid meta-heuristic algorithm for optimal design of large-scale dome structures, Eng Optim 2017; 49(4): 1–18. doi:10.1080/0305215X.2017.1313250.

17. Erol OK, Eksin I. A new optimization method: Big Bang–Big Crunch, Adv Eng Softw 2006; 37(2): 106-11. doi:10.1016/j.advengsoft.2005.04.005.
18. Kaveh A, Talatahari S. Size optimization of space trusses using Big Bang–Big Crunch algorithm, Comput Struct 2009; 87(17–18): 1129–40. doi:10.1016/j.compstruc.2009.04.011.
19. Rashedi E, Nezamabadi-pour H, Saryazdi S. GSA: A gravitational search algorithm, Inform Sci 2009; 179(13): 2232–48. doi:10.1016/J.INS.2009.03.004.
20. Kaveh A, Kalateh-Ahani M, Masoudi MS. the Cma Evolution Strategy Based Size Optimization of Truss Structures, Int J Optim Civil Eng 2011; 1(2): 233-56. http://ijoce.iust.ac.ir/browse.php?a_id=17&sid=1&slc_lang=enAccessed.
21. Kaveh A, Mahdavi VR. Colliding Bodies Optimization method for optimum discrete design of truss structures, Comput Struct 2014; 70(1): 1–12. doi:10.1016/j.compstruc.2014.04.006.
22. Kaveh A, Ilchi Ghazaan M. Enhanced colliding bodies optimization for design problems with continuous and discrete variables, Adv Eng Softw 2014; 77: 66–75. doi:http://dx.doi.org/10.1016/j.advengsoft.2014.08.003.
23. Kaveh A, Talatahari S. A novel heuristic optimization method: charged system search, Acta Mech 2010; 3(10). doi:10.1007/s12192-010-0223-9.
24. Kaveh A, Talatahari S. Optimal Design of Skeletal Structures via the Charged System Search Algorithm, 2010: 893–911. doi:10.1007/s00158-009-0462-5.
25. Kaveh A, Motie Share MA, Moslehi M. Magnetic charged system search: A new meta-heuristic algorithm for optimization, Acta Mech 2013; 224(1): 85–107. doi:10.1007/s00707-012-0745-6.
26. Kaveh A, Mirzaei B, Jafarvand A. An improved magnetic charged system search for optimization of truss structures with continuous and discrete variables, Appl Soft Comput J 2015; 28: 400–10. doi:10.1016/j.asoc.2014.11.056.
27. Kaveh A, Khayatazad M. A new meta-heuristic method: Ray Optimization, Comput Struct 2012; 112–113(10): 283–294. doi:10.1016/j.compstruc.2012.09.003.
28. Hatamlou A. Black hole: A new heuristic optimization approach for data clustering, Inform Sci 2013; 222:175-84. doi:10.1016/J.INS.2012.08.023.
29. Formato RA. Central force optimization: a new metaheuristic with applications in applied electromagnetics, Progr Electromag Res 2007; 77: 425–491. doi:10.2528/PIER07082403.
30. Moghaddam FF, Moghaddam RF, Cheriet M. Curved Space Optimization: A Random Search based on General Relativity Theory, 2012. https://arxiv.org/abs/1208.2214AccessedDecember 11, 2018.
31. Du H, Wu X, Zhuang J. Small-World optimization algorithm for function optimization, In: International Conference on Advances in Natural Computation, Springer; 2006: pp. 264–273.
32. Alatas B. ACROA: Artificial Chemical Reaction Optimization Algorithm for global optimization, Expert Syst Applicat 2011; 38(10): 13170–80. doi:10.1016/J.ESWA.2011.04.126.
33. Kaveh A, Bakhshpoori T. A new metaheuristic for continuous structural optimization: water evaporation optimization, Struct Multidisc Optim 2016; 54(1): 23–43. doi:10.1007/s00158-015-1396-8.

34. Hosseini HS, Hamed. Principal components analysis by the galaxy-based search algorithm: a novel metaheuristic for continuous optimisation, Int J Computat Sci Eng 2011; 6(1/2): 132. doi:10.1504/IJCSE.2011.041221.
35. Kennedy J, Eberhart R. Particle swarm optimization, Neural Networks, 1995, Proceedings IEEE International Conference on 1995; 4: pp. 1942–1948. doi:10.1109/ICNN.1995.488968.
36. Kaucic M, Barbini F, Camerota Verdù FJ. Polynomial goal programming and particle swarm optimization for enhanced indexation, Soft Comput 2019; (2009), doi:10.1007/s00500-019-04378-5.
37. Gomes HM. Truss optimization with dynamic constraints using a particle swarm algorithm, Expert Syst Applicat 2011; 38(1): 957–968. doi:10.1016/j.eswa.2010.07.086.
38. Li LJ, Huang ZB, Liu F. A heuristic particle swarm optimization method for truss structures with discrete variables, Comput Struct 2009; 87(7–8): 435–43. doi:10.1016/j.compstruc.2009.01.004.
39. Yang XS, Suash Deb. Cuckoo Search via Lévy flights, In: 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), IEEE; 2009: pp. 210–214.
40. Gandomi AH, Talatahari S, Yang X-S. Design optimization of truss structures using cuckoo search algorithm, The Struct Des Tall Special Build 2013; 22(June 2012): 1330–49. doi:10.1002/tal.
41. Zelinka I, Bukacek M. Gold rush - A swarm dynamics in games, AIP Conference Proceedings 2017; 1863: pp. 1–5. doi:10.1063/1.4992253.
42. Tejani G, Savsani VJ, Mirjalili S, Patel VK. Truss optimization with natural frequency bounds using improved symbiotic organisms search, Knowl-Based Syst 2018; 143:162–78. doi:10.1016/j.knosys.2017.12.012.
43. Isaac B, Allaire D. Expensive Black-box model optimization via a gold rush policy, J Mech Des 2019; 141(March). doi:10.1115/1.4042113.
44. Mirjalili SA, Lewis A. The Whale Optimization Algorithm, *Adv Eng Softw* 2016; **95**: 51-67. doi:10.1016/j.advengsoft.2016.01.008.
45. Kaveh A, Farhoudi N. A new optimization method: Dolphin Echolocation, *Adv Eng Softw* 2013; **59**(3): 53-70. doi:10.1016/j.advengsoft.2013.03.004.
46. Dorigo M, Maniezzo V, Colorni A. Ant system: optimization by a colony of cooperating agents. *IEEE Transact Syst, Man Cybernet, Part B (Cybernetics)* 1996; **26**(1): 29–41. doi:10.1109/3477.484436.
47. Abbass HA. MBO: marriage in honey bees optimization-a Haplometrosis polygynous swarming approach. In: *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*.
48. Degertekin SO, Hayalioglu MS. Sizing truss structures using teaching-learning-based optimization, *Comput Struct* 2013; **119**: 177-88. doi:10.1016/j.compstruc.2012.12.011.
49. Rao RV. *Teaching Learning Based Optimization Algorithm and Its Engineering Applications*, 2016.
50. Zong Woo Geem ZW, Joong Hoon Kim JH, Loganathan GV. A New Heuristic Optimization Algorithm: Harmony Search, *Simulat* 2001; **76**(2): 60-8. doi:10.1177/003754970107600201.

51. Glover F. Tabu search - Part I, *ORSA J Comput* 1989; **1**: 190–206. doi:10.1007/978-1-4419-7997-1_17.
52. He S, Wu QH, Saunders JR. Group search optimizer: An optimization algorithm inspired by animal searching behavior, *IEEE Transact Evolut Computat* 2009; **13**(5): 973-90. doi:10.1109/TEVC.2009.2011992.
53. Mucherino A, Seref O, Seref O, Kundakcioglu OE, Pardalos P. Monkey search: a novel metaheuristic search for global optimization. In: *AIP Conference Proceedings* 2007; AIP; **953**: pp. 162–173.
54. Shiqin Y, Jianjun J, Guangxing Y. A Dolphin Partner Optimization, In: *2009 WRI Global Congress on Intelligent Systems*, *IEEE*; 2009: pp.124–128.
55. Lu X, Zhou Y. A Novel Global Convergence Algorithm: Bee Collecting Pollen Algorithm. In: *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*. Berlin, Heidelberg: Springer Berlin Heidelberg; 2008: pp. 518–525.
56. Mirjalili SA, Mirjalili SM, Lewis A. Advances in Engineering Software Grey Wolf Optimizer, *Adv Eng Softw* 2014; **69**: 46–61. doi:10.1016/j.advengsoft.2013.12.007.
57. Reeves K, Frost L, Fahey C. Integrating the historiography of the nineteenth-century gold rushes, *Australian Econom History Rev* 2010; **50**(2): 111-28. doi:10.1111/j.1467-8446.2010.00296.x.
58. Elliott L. Mancall, David G. Brock. *Gray's Clinical Neuroanatomy*, US Elsevier Health Bookshop; 2012.
59. Kamalinejad M, Arzani H, Kaveh A. Quantum evolutionary algorithm with rotational gate and H -gate updating in real and integer domains for optimization, *Acta Mech* 2019: 1–25. doi:10.1007/s00707-019-02439-2.
60. Mirjalili SA, Mirjalili SM, Hatamlou A. Multi-Verse Optimizer: a nature-inspired algorithm for global optimization, *Neural Comput Applicat* 2016; **27**(2): 495–513. doi:10.1007/s00521-015-1870-7.
61. Askarzadeh A. A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm, *Comput Struct* 2016; **169**: 1–12. doi:10.1016/j.compstruc.2016.03.001.
62. Chickermane H, Gea HC. Structural optimization using a new local approximation method, *Numer Meth Eng* 1996; **39**(August 1994): 829–46.
63. Gandomi AH, Yang X. Cuckoo search algorithm : a metaheuristic approach to solve structural optimization problems, *Eng Comput* 2013; **29**(1): 17-35. doi:10.1007/s00366-011-0241-y.
64. Cheng M, Prayogo D. Symbiotic Organisms Search : A new metaheuristic optimization algorithm. *Comput Struct* 2014; **139**:98–112. doi:10.1016/j.compstruc.2014.03.007.
65. Belegundu AD, Arora JS. A study of mathematical programming methods for structural optimization. Part I: Theory, *Int J Numer Meth Eng* 1982; **21**(9): 1583-99. doi:10.1002/nme.1620210904.
66. Arora JS. *Introduction to Optimum Design*, Elsevier/Academic Press, 2004.
67. Sadollah A, Bahreininejad A, Eskandar H, Hamdi M. Mine blast algorithm : A new population based algorithm for solving constrained engineering optimization problems, *Appl Soft Comput J* 2013; **13**(5): 2592–2612. doi:10.1016/j.asoc.2012.11.026.

68. Kaveh A, Motie Share MA, Moslehi M. Magnetic charged system search: A new meta-heuristic algorithm for optimization, *Acta Mech* 2013; **224**(1): 85–107. doi:10.1007/s00707-012-0745-6.
69. Rao S. *Engineering Optimization: Theory and Practice: Fourth Edition*, John Wiley & Sons, 2009.
70. Kaveh A. Ataollah Zaerreza. Shuffled shepherd optimization method a new Meta-heuristic algorithm, *Eng Computat* 2020; ahead-of-print. doi:10.1108/EC-10-2019-0481.
71. Sandgren E. Nonlinear integer and discrete programming in mechanical design optimization, *J Mech Des* 1990; **112**(2): 223. doi:10.1115/1.2912596.
72. Hasançebi O, Çarbaş S, Doğan E, Erdal F, Saka MP. Performance evaluation of metaheuristic search techniques in the optimum design of real size pin jointed structures, *Comput Struct* 2009; **87**(5–6): 284–302. doi:10.1016/j.compstruc.2009.01.002.
73. Kaveh A, Zakian P. Enhanced bat algorithm for optimal design of skeletal structures, *Asian J Civil Eng* 2014; **15**(2): 179–212.
74. Aslani M, Ghasemi P, Gandomi AH. Constrained mean-variance mapping optimization for truss optimization *problems*, *Struct Des Tall Special Build* 2018; **27**(6): 1–17. doi:10.1002/tal.1449.
75. Kaveh A, Sheikholeslami R, Talatahari S, Keshvari-Ilkhichi M. Chaotic swarming of particles: A new method for size optimization of truss structures, *Adv Eng Softw* 2014; **67**: 136–47. doi:10.1016/j.advengsoft.2013.09.006.
76. Kaveh A, Massoudi MS. Multi-objective optimization of structures using Charged System Search, *Sci Iran* 2014; **21**(6): 1845-60.
77. Sonmez M. Performance comparison of metaheuristic algorithms for the optimal design of space trusses, *Arabian J Sci Eng* 2018; **43**(10): 5265-5281. doi:10.1007/s13369-018-3080-y.
78. American Institute of Steel Construction (AISC). Manual of steel construction allowable stress design (1989), 9th ed, Chicago, IL.
79. Kaveh A, Mahdavi VR. Colliding bodies optimization method for optimum discrete design of truss structures, *Comput Struct* 2014; **139**: 43–53. doi:10.1016/j.compstruc.2014.04.006.